

REMARKS

This Amendment is submitted in response to the Office Action dated April 8, 2005. Claims 1, 7, and 13 have been amended, and claims 19-21 have been added. Claims 1-21 are presently pending.

CLAIMS INTERPRETATION

On page 2 of the Office Action, it is indicated that the phrase "instrumentation event" is interpreted as "debug event" or "debug probe." Applicants contend that as explained throughout the specification and in particular at ..., an "event" is properly interpreted as conditions or events occurring within a simulation model that are detectable by HDL logic constructs. (see specification page 29, lines 14-19). The exemplary "instrumentation events" described in the specification include fail, count, and harvest events.

CLAIM REJECTIONS UNDER 35 U.S.C. § 103

In the present Office Action, Claims 1-4, 7-10, and 13-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,011,920, issued to Edwards et al. (hereinafter referred to as *Edwards*) and U.S. Patent No. 6,336,087, issued to Burgun et al., (hereinafter referred to as *Burgun*).

Regarding the grounds for rejecting claims 1, 7, and 13, in item 3 on page 4 of the Office Action, Applicants disagree that *Edwards* discloses assembling or in any way storing or maintaining an "instrumentation event disable list" (i.e., list of event items to be disabled). Applicants agree that *Edwards* discloses an application list listing "known" applications that need to be debugged, but note that this is not a list of events to be disabled, and is therefore not the same or equivalent to an "event disable list" as recited in Applicants' claims. (See col. 2, lines 35-38, explaining list of applications constructed by polling Microkernel loaders). Furthermore, the applications listed by *Edwards* are not equivalent to instrumentation "events," clearly defined in Applicants' specification as explained above. In contrast to listing "events" to be disabled, *Edward's* "application list" signifies the gathering in a distributed environment of one or more applications that are subject to being debugged (see col. 2, lines 23-14; col. 3, lines 63-67). Comparing the application debugging context of *Edwards* with the simulation model testing context of Applicants' proposed system, the closest analogy to *Edwards'* "application list" would be a list

of simulation models, since it is the models themselves, and not the “events” occurring during simulation testing, which are being tested.

In further regard to item 3 on page 4 of the Office Action, Applicants disagree that *Edwards* teaches:

“and prior to simulating said simulation model within said simulation client:
retrieving said instrumentation event disable list from said instrumentation server;
and
disabling instrumentation events specified within said instrumentation event disable list.”

Nothing in *Edwards* relates to simulating or otherwise processing simulation models, or maintaining/processing an type of “disable list” as explained above.

The paragraph beginning on page 5, line 12 of item 3 of the Office Action asserts that *Burgun* teaches disabling/tracking an instrumentation event in a simulation model where the simulation client communicates with an instrumentation server to process simulation data with respect to the simulation model (Fig. 21). In Fig. 21 and elsewhere, *Burgun* includes disclosure relating to processing of “simulation models” as that term is used in Applicants’ specification and claims. However, the processing relating to the “instrumentation” is different than Applicants’ proposed invention. Namely, *Burgun* discloses synthesizing RTL source code to generate instrumentation logic corresponding to RTL source code statements. As with *Edwards*, nothing in *Burgun* discloses any method for maintaining an “event disable list” or any other system or technique for disabling simulation events in any manner.

Regarding the paragraph beginning on page 6, line 9 of item 3 of the Office Action, Applicants contend that the disclosure of *Burgun* is not properly combinable with *Edwards*’s disclosure (or vice versa), since these references are directed to fundamentally unrelated concepts as explained above.

The foregoing traversals notwithstanding, Applicants have amended independent claims 1, 7, and 13 and have added independent claim 19 and dependent claims 20-21, to more specifically characterize and distinguish the proposed invention from the subject matter disclosed by *Edwards* and *Burgun*. Namely, claim 1 is now specifically directed to an HDL simulation event context and more specifically characterizes the natures of an “instrumentation event” and the “disabling” of such an event as constituting an effective “masking” rather than a system

interruption (e.g. a breakpoint). Specifically, claim 1 (and similarly for claims 7 and 13) now recites a method for disabling monitoring of an instrumentation event in a hardware description language (HDL) model of a circuit design within a batch simulation farm, said HDL model including an instrumentation entity that detects an instrumentation event and delivers a signal to an instrumentation logic block to indicate an occurrence of an instrumentation event during simulation of said circuit design, said batch simulation farm including a simulation client in which the HDL model is simulation tested and further including an instrumentation server communicatively coupled to the simulation client, said method comprising..." (Emphasis added). The body of claim 1 has also been amended to recite:

"assembling an instrumentation event disable list within said instrumentation server, wherein said instrumentation event disable list lists instrumentation events to be disabled within said HDL simulation model; and

prior to simulating said HDL model within said simulation client:

retrieving said instrumentation event disable list from said instrumentation server;

and

disabling monitoring of instrumentation events specified by said instrumentation event disable list." (Emphasis added).

Newly added independent claim 19 substantially incorporates much of the substance of independent claim 1 and includes further limitations relating to the manner of centrally maintaining "event disable lists." Specifically, the method of claim 19 expressly requires that an event disable list be maintained for each of potentially multiple simulation models in a batch simulation farm. Namely, claim 19 includes steps of, "maintaining, within said instrumentation server, an event disable list for each active HDL model within said simulation batch farm," and "including within each event disable list, a list of instrumentation events to be disabled for a corresponding HDL model" (emphasis added). No new matter has been added in amending claims 1, 7, and 13 or in adding claims 19-21.

For the foregoing reasons, Applicants contend that neither *Edwards* nor *Burgun*, individually or in combination, disclose or suggest the foregoing discussed elements of amended claims 1, 7, and 13 and newly added claim 19 and therefore do not anticipate or render these claims obvious. Therefore, Applicants submit that claims 1, 7, 13, 19, and all claims depending therefrom are in condition for allowance and a notice to that effect is requested.

Applicants have diligently responded by particularly pointing out how the pending claims 1-21 are patentably distinct from the prior art references cited in the foregoing claim rejections. Applicants believe claims 1-21 are in condition for allowance and a notice to that effect is respectfully requested.

Applicants invite the Examiner to contact the undersigned attorney of record at 512.343.6116 if such would further or expedite the prosecution of the present Application.

Respectfully submitted,



Matthew W. Baca
Reg. No. 42,277
Dillon & Yudell LLP
8911 North Capital of Texas Highway
Suite 2110
Austin, Texas 78759
512.343.6116

ATTORNEY FOR APPLICANT(S)